

API FIRMAO

Table of contents

[1. API Enablement and Authentication](#)

[2. Adding new objects \(POST\)](#)

[a\) Projects](#)

[b\) Tasks](#)

[c\) Products/Services](#)

[d\) Counterparty](#)

[e\) Transactions](#)

[f\) Offers/Orders/Contracts](#)

[g\) Contact persons](#)

[h\) Trading notes](#)

[i\) Sales opportunities](#)

[3. Object modification \(PUT\)](#)

[a\) Change of product name:](#)

[b\) Change of the contractor's name:](#)

[c\) Changing the product inventory \(required to enable the organization's setting: Direct modification of stock levels\)](#)

[4. Object Fetching \(GET\)](#)

[Paging](#)

[Collation](#)

[Filtering](#)

[Response codes](#)

[a\) Downloading projects](#)

[b\) Downloading tasks](#)

[c\) Downloading products](#)

[d\) Downloading customers](#)

- [e\) Downloading transactions](#)
- [f\) Downloading transactions for a counterparty](#)
- [g\) retrieving transaction items - transactions](#)
- [h\) downloading offers/orders](#)
- [i\) download of sales opportunities](#)
- [j\) Getting contacts](#)
- [k\) downloading customergroups](#)
- [l\) retrieving sales notes - salesnotes](#)
- [m\) downloading warehouse documents - storedocuments](#)
- [n\) download the list of files - documents](#)
- [o\) downloading mails - mails](#)
- [p\) downloading a series of numbering invoices](#)

[5. Deleting objects \(PUT, DELETE\)](#)

[6. A detailed description of creating, editing, finishing, and starting work times](#)

[7. PDF Download](#)

- [a\) Invoices](#)
- [b\) Offers/orders/contracts](#)

[8. Uploading files/attachments to objects](#)

- [a\) \(tasks/projects/products/services/customers/offers/orders/sales notes/opportunities\)](#)

[9. Response Codes](#)

[10. Sample php scripts](#)

- [a\) adding a contractor](#)
- [b\) downloading the contractor's data](#)
- [c\) adding transactions](#)
- [d\) Uploading the PHP file < 7.0](#)
- [e\) uploading php file >= 7.0](#)

[11. More API Options](#)

1. API Enablement and Authentication

For the API to work, you need to go to the Company->Settings->API settings panel, turn on the API, download the login and generate a password.

All requests must have authentication in the form of:

"Authorization" header with the value "Basic " + base64 encoded string username + ":" + password

example:

Authorization: Basic ai5zbWI0aEBzdGVsbGlzLmJpczoxMjM0NTY=

where ai5zbWI0aEBzdGVsbGlzLmJpczoxMjM0NTY is the base64 encoded string "j.smith@stellis.bis:123456"

username - login of the firmao system API user

password - password of the API user of the firmao system

The API allows up to 60 requests per minute, with a maximum of 5 concurrent requests at any given time. Exceeding either limit will result in an HTTP 429 (Too Many Requests) response.

2. Adding new objects (POST)

The creation of new objects is done using a POST request and should be directed to

`https://system.firmao.pl/{OrganizationID}/svc/v1/{xxx}xxx` - name of the object type

OrganizationIdentifier - organization identifier visible in the url after logging in to the company

Data type:

ContentType = application/json; charset=UTF-8

All fields available in the objects are described in the data import help.

Custom fields have the keys "customFields.custom5" and accept Strings. Available fields can also be "previewed" by retrieving the data using the GET method (excluding the permission field, which is enumerated and cannot be provided in the POST/PUT request).

In response to POST requests, we send data in the form of a json with information about all fields that have been calculated/changed by firmao. Among other things, the

id of the created object [{"modification":"CREATE" "objectId":"50007" }]

Response codes: The API uses standard HTTP error codes.

201 - correct creation of 4xx and 5xx - an error occurred during the creation of the object. In the case of errors, a clear description of the error appears as standard. For example, if you try to create a task for a non-existent project, you will get:Response code: 400.

```
{"errorCode":"entityNotFoundForId","field":"project","rejectedValue":"50","additionalInfo":{"id":"50","cls":"project"}}
```

The author field will always show the API user.

Sample requests

a) Projects

<https://system.firmao.pl/{OrganizationID }/svc/v1/projects>

```
POST{  
  
    "description":"Description",  
  
    "teamMembers":[{"id":6}],
```

```
"restrictedTeamMembers":[]

,"managers":[{"id":10}]

,"tags":[{"id":3}],

"startDate":"2012-07-17T00:00:00+02:00",

"name":"Example"

}
```

Response if the formation is correct

Code 201:

```
{"changelog":[{"deletedWith":"","modification":"CREATE","newValuesDictionary":false,"objectClass":"project","objectId":50003,"objectLabel":"Example","oldValuesDictionary":false,"property":""}]}
```

b) Tasks

<https://system.firmao.pl/{OrganizationID}/svc/v1/tasks>

POST

```
{

"taskType":"PROJECT",

"priority":"300",
```

```
"tags":[],
"responsibleUsers":[{"id":2}],
"estimatedHours":5,
"plannedEndDateType":"EXACTLY"
,"plannedStartDateType":"EXACTLY"
,"description":"Description",
"plannedStartDate":"2012-07-17T00:00:00+02:00",
"plannedEndDate":"2012-07-18T00:00:00+02:00",
"orderCalculateFromGross":false,
"orderVatPercent":null,
"name":"Example task",
"status":"WAITING"
}
```

Response if the formation is correct

Code 201:

```
{"changelog":[{"deletedWith":"","modification":"CREATE","newValuesDictionary":false,"objectClass":"task","objectId":50017,"objectLabel":"Sample task","oldValuesDictionary":false,"property":""},{deletedWith":"","modification":"CHANGE","newValueClass":"date","newValueDisplay":"2012-07-17T14:00:00.000+02:00","new
```

```
ValuesDictionary":false,"objectClass":"task","objectId":50017,"objectLabel":"Example  
task","oldValueClass":"date","  
oldValueDisplay":"2012-07-17T22:00:00.000Z","oldValuesDictionary":false,"property":"p  
lannedEndDate"]}]}
```

c) Products/Services

<https://system.firmao.pl/{OrganizationID}/svc/v1/products>

POST

```
{  
  
  "tags":[],  
  
  "purchasePriceGroup.netPrice":10,  
  
  "purchasePriceGroup.grossPrice":12.3,  
  
  "purchasePriceGroup.vat":23,  
  
  "saleable":true,  
  
  "productCode":"PR1",  
  
  "name":"Sample Product"  
}
```

Response if the formation is correct

Code 201:

```
{"changelog":[{"deletedWith":"","modification":"CREATE","newValuelsDictionary":false,"objectClass":"product","objectId":50001,"objectLabel":"Sample Product","oldValuelsDictionary":false,"property":""}]}
```

d) Counterparty

POST

<https://system.firmao.pl/{OrganizationID}/svc/v1/customers>

```
{ "bankAccountNumber" : "50 1020 5558 1111 1702 0600 0067",  
  
  "correspondenceAddress.city" : "Pruszków",  
  
  "correspondenceAddress.country" : "United States",  
  
  "correspondenceAddress.county" : "",  
  
  "correspondenceAddress.postCode" : "05-800",  
  
  "correspondenceAddress.street" : "Stefana Batorego 3/4",  
  
  "customerGroups" : [ ],  
  
  "customerType" : "PARTNER",
```

"description" : "A company specializing in computer service",

"emails" : ["kompSERVICE.firmao@mailinator.com",

 "komputery@mailinator.com"

],

"employeesNumber" : 1,

"faxNumber" : "22 398 78 87",

"label" : "KOMP-SERVICE",

"name" : "KOMP-SERVICE Jan Kowalski",

"nipNumber" : "7291077843",

"officeAddress.city" : "Pruszków",

"officeAddress.country" : "Poland",

"officeAddress.postCode" : "05-800",

"officeAddress.street" : "Stefana Batorego 3/4",

"ownership" : "PRIVATE",

"phones" : ["223457426"]

"status" : null,

"tags" : [],

"voivodeship" : null,

"website" : "www.kompSERVICE.bis"

```
}
```

Response if the formation is correct

Code 201:

```
{"changelog":[{"deletedWith":"","modification":"CREATE","newValuesDictionary":false,"objectClass":"customer","objectId":50001,"objectLabel":"KOMP-SERVICE","oldValuesDictionary":false,"property":""}]}
```

Reply for a repeated field (ID or NIP number):

Code 409:

```
{"errorCode":"duplicateValueFoundException","objectName":"customer","fieldName":"nipNumber","duplicateValues":[{"lineNumber":-1,"objectId":51,"value":"7291077843"}]}
```

e) Transactions

POST

<https://system.firmao.pl/{identyfikatorOrganizacji}/svc/v1/transactions>

```
{
```

```
"transactionEntries":[
```

```
{  
  "classificationNumber":"","  
  "discount":null,  
  "productCode":"ST0003",  
  "quantity":1,  
  "unit":null,  
  "vatPercent":0,  
  "product":52, if we provide text (string) instead of a number, an entry with  
this text will be created that is not associated with any product  
  "unitNettoPrice":"1000.00"  
}  
],  
"transactionNettoPrice":"1000.00",  
"transactionBruttoPrice":"1000.00",  
"transactionVatPrice":"0.00",  
"invoiceAnnotations":null,  
"paid":false,  
"currency":"PLN",  
"paymentType":"CASH",
```

```
"restOfPaidValue":1000,

"paidValue":0,

"invoiceDate":"2012-08-08",

"paymentDate":"2012-08-15",

"transactionDate":"2012-08-08",

"type":"INVOICE",

"mode":"SALE",

"createNewCustomerIfNeeded":true, //If createNewCustomerIfNeeded is true, a
new customer will be created with further customerName, phones and emails
parameters. If a counterparty with such a name already exists, it will be linked to
the transaction

"customerName":"test",

"phones":[ Use only if createNewCustomerIfNeeded: true

    "111222333"

],

"emails":[ Use only if createNewCustomerIfNeeded: true

    "test@test.test"

],

"customerAddress.postCode":"91-477",

"customer":50,
```

```
"customerAddress.city":"Boat",  
  
"customerAddress.country":"United States",  
  
"customerAddress.street":"Piotrkowska 147/8",  
  
"nipNumber":"1060000062",  
  
"issuingPerson":"Jacob Smith",  
  
"transactionNumber":"1/8/2012" We can also set  
  
"autoNumber":true if we want the numbering to be automatic.  
  
}
```

Response if the formation is correct

Code 201:

```
{"changelog":[{"deletedWith":"","modification":"CREATE","newValuesDictionary":false,"objectClass":"transaction","objectId":50035,"objectLabel":"1/8/2012 - Internet Center","oldValuesDictionary":false,"property":""},{deletedWith":"","modification":"CREATE","newValuesDictionary":false,"objectClass":"commisionuser","objectId":35,"objectLabel":"","oldValuesDictionary":false,"property":""},{deletedWith":"","modification":"CHANGE","newValueClass":"long","newValueDisplay":"10","newValuesDictionary":false,"objectClass":"customer","objectId":50,"objectLabel":"Internet Center","oldValueClass":"long","oldValueDisplay":"9","oldValuesDictionary":false,"property":"transactionsCount"}], "showCustomVatWarning":"true"}
```

If a sales transaction with a given number already exists, the transaction is not created and in response we get:

Code 400:

```
{"errorCode":"transactionNumberAlreadyExists","additionalInfo":{"suggestedNumber":"2/8/2012"}}
```

For a **credit memo**, the items must contain two entries for each row in the original document: the line item with the values before the correction and the line item with the values after the correction.

Example of a correction invoice for the above transaction:{

"invoicePatternId": 1,

"exchangeRate": 1,

"exchangeRateDate": "2012-08-07",

"factor": 1,

"correctionCause": "Incorrectly entered data.",

"paid": false,

"bankAccount": 1,

"currency": "PLN",

"paymentType": "CASH",

"restOfPaidValue": 1000,

"paidValue": 0,
"invoiceDate": "2022-04-08",
"invoiceReceptionDate": "2022-04-08",
"paymentDate": "2022-04-15",
"transactionDate": "2012-08-08",
"type": "CORRECTION",
"mode": "SALE",
"customerAddress.postCode": "91-477",
"customer": 50,
"customerAddress.country": "United States",
"customerAddress.city": "Boat",
"customerAddress.street": "Piotrkowska 147/8",
"nipNumber": "106-00-00-062",
"recipientPhoneNumber": null,
"connectedDocDate": "2012-08-08", // **date of the original document**
"issuingPerson": "Jacob Smith",
"splitPayment": false,
"transactionNumber": "2022/04/004/KOR",
"connectedDocNumber": "1/8/2012", // **original document number**

```
"calculateFromGross": false,

"transactionEntries": [

    { // position before

      "unit": null,

      "classificationNumber": "",

      "baseEntryId": 50106, // the item of the original document to be corrected

      "productCode": "ST0003",

      "quantity": 1,

      "entryPaid": "NOT_PAID",

      "entrySize": 1,

      "variant": 1,

      "vatPercent": "0",

      "recordId": 16, // a number identifying the position within the document.

      "salePriceGroup": "A",

      "unitPurchaseNettoPrice": 0,

      "unitNettoPrice": "1000.00",

      "unitNettoPriceWithoutDiscount": "1000.00",

      "product": 52

    },
```

```
    { // "after" position

    "unit": null,

    "classificationNumber": "",

    "productCode": "ST0003",

    "quantity": 2,

    "entryPaid": "NOT_PAID",

    "entrySize": 1,

    "variant": 1,

    "vatPercent": 0,

    "recordId": 17, // number identifying the position within the document.

    "salePriceGroup": "A",

    "unitPurchaseNettoPrice": 0,

    "unitNettoPrice": "1000.00",

    "unitNettoPriceWithoutDiscount": "0.00",

    "invalidRecordId": 16, // recordId of the "before" position

    "product": 52

    }

  ],

  "transactionNettoPrice": "1000.00",
```

```
"transactionBruttoPrice": "1000.00",  
  
"transactionVatPrice": "0.00",  
  
"lastSelectedInvoiceReport": "Faktura_korygujaca2"  
  
}
```

f) Offers/Orders/Contracts

POST

<https://system.firmao.pl/{OrganizationID}/svc/v1/offers> (Offers / Orders)

<https://system.firmao.pl/{OrganizationID}/svc/v1/agreements> (Agreements)

```
{  
  
  "entries":[ entries entry can be omitted if there are no bid items  
  
    {  
  
      "unit":"pcs",  
  
      "product":51, if we provide a text (string) instead of a number, an entry with  
this text will be created that is not associated with any product. If you enter a  
number, the system searches for an existing product and attaches it to the item.  
  
      "classificationNumber": "",
```

```
"productCode":"ST0002",  
  
"quantity":1,  
  
"vatPercent":22,  
  
"unitNettoPrice":"818.85"  
  
}  
  
],  
  
"nettoPrice":"818.85",  
  
"grossPrice":"999.00",  
  
"vatPrice":"180.15",  
  
"currency":"PLN",  
  
"paymentType":"CASH",  
  
"paymentDate":"2012-11-09",  
  
"offerDate":"2012-11-09",  
  
"validFrom":"2012-11-09",  
  
"type":"OFFER", OFFER - for the offer, ORDER - for the order, AGREEMENT - for  
the agreement,  
  
"mode":"SALE", SALE - for sales, PURCHASE - for purchase  
  
"offerStatus":"NEW", NEW, SENT, DURING_NEGOTIATIONS, ACCEPTED,  
REJECTED, EXECUTED
```

```
"customerAddress.postCode":"91-477",  
  
"daysToDueDate":7,  
  
"customer":50, the ID of the customer in the system - you can use the  
customerName parameter if you do not want to link it to the customer  
  
"customerAddress.city":"Boat",  
  
"customerAddress.country":"United States",  
  
"customerAddress.street":"Piotrkowska 147/8",  
  
"nipNumber":"1060000062",  
  
"issuingPerson":"Jacob Smith",  
  
"number":"1/11/2012/U" required field and must be unique  
  
}
```

Response if the formation is correct

Code 201:

```
{"changelog":[{"deletedWith":"","modification":"CREATE","newValuesDictionary":false,"objectClass":"offer","objectId":2,"objectLabel":"1/11/2012/U - Internet Center","oldValuesDictionary":false,"property":""},{ "deletedWith":"","modification":"CHANGE","newValueClass":"long","newValueDisplay":"2","newValuesDictionary":false,"objectClass":"customer","objectId":50,"objectLabel":"Internet Center","oldValueClass":"long","oldValueDisplay":"1","oldValuesDictionary":false,"property":"offersCount"}]}
```

g) Contact persons

POST

<https://system.firmao.pl/{identyfikatorOrganizacji}/svc/v1/contacts>

```
{  
  
  "firstName":"Marzena",  
  
  "lastName":"Wojciechowska",  
  
  "position":"Salesperson",  
  
  "description":null,  
  
  "emails":[  
  
    "m.wojciechowska@mailinator.com"  
  
  ],  
  
  "phones":[  
  
    "226548484",  
  
    "226548485",  
  
    "500111222"
```

```

], "tags":[

],

"customer":51, customer id in the system - Required field

"label":"Marzena Wojciechowska"

}

```

Response if the formation is correct

Code 201:

```

{"changelog":[{"deletedWith":"","modification":"CREATE","newValuesDictionary":false,"objectClass":"contact","objectId":50001,"objectLabel":"Marzena Wojciechowska","oldValuesDictionary":false,"property":""},{deletedWith":"","modification":"CHANGE","newValueClass":"long","newValueDisplay":"0","newValuesDictionary":false,"objectClass":"customer","objectId":51,"objectLabel":"KOMP-SERVICE","oldValueClass":"long","oldValueDisplay":"0","oldValuesDictionary":false,"property":"mailsCount"},{"deletedWith":"","modification":"CHANGE","newValueClass":"long","newValueDisplay":"2","newValuesDictionary":false,"objectClass":"customer","objectId":51,"objectLabel":"KOMP-SERVICE","oldValueClass":"long","oldValueDisplay":"0","oldValuesDictionary":false,"property":"contactsCount"},{"deletedWith":"","modification":"ATTACH","newValueClass":"contact","newValueDisplay":"Marzena Wojciechowska","newValueId":"50001","newValuesDictionary":false,"objectClass":"customer","objectId":51,"objectLabel":"KOMP-SERVICE","oldValuesDictionary":false,"property":"contact_change"}]}

```

h) Trading notes

POST

<https://system.firmao.pl/{OrganizationID}/svc/v1/salesnotes>

```
{  
  
  "description":"Test note",  
  
  "type":"NOTE",  
  
  "customer":51, customer id in the system - Required field  
  
  "tags":[  
  
  ],  
  
  "date":"2013-04-10T17:12:20+02:00"  
  
}
```

Response if the formation is correct

Code 201:

```
{"changelog":[{"deletedWith":"","modification":"CREATE","newValuesDictionary":false,"objectClass":"salesnote","objectId":50011,"objectLabel":"","oldValuesDictionary":false,"pr
```

```
property":"","{"deletedWith":"","modification":"CHANGE","newValueClass":"long","newValueDisplay":"1","newValuesDictionary":false,"objectClass":"customer","objectId":51,"objectLabel":"KOMP-SERVICE","oldValueClass":"long","oldValueDisplay":"0","oldValuesDictionary":false,"property":"salesNotesCount"},"{"deletedWith":"","modification":"CHANGE","newValueClass":"date","newValueDisplay":"2017-09-21T13:49:11.691Z","newValuesDictionary":false,"objectClass":"customer","objectId":51,"objectLabel":"Name","oldValuesDictionary":false,"property":"lastContactDate"]}]}
```

i) Sales opportunities

POST

<https://system.firmao.pl/{OrganizationID}/svc/v1/salesopportunities>

```
{
  "customer":1,
  "observers":[],
  "responsibleUser":1,
  "acquisitionMethod":"MARKETING_CAMPAIGN",
  "sellingProcess":"DEFAULT",
  "salesDate":"2023-09-14",
  "salesOpportunityValue":11111
  , "tags":[],
```

```

"emails":[
    "dell.firmao@mailinator.com"
],
"phones":[
    "+48123123123"
],
"label":"Test Name", Opportunity Name - Required Field
"currency":"PLN", Currency - Required field
"stage":"2"
}

```

Response if the formation is correct

Code 201:

```

{"changelog":[{"deletedWith":"","modification":"CREATE","newValuesDictionary":false,"objectClass":"salesopportunity","objectId":8,"objectLabel":"test - DELL","oldValuesDictionary":false,"property":""},{ "deletedWith":"","modification":"CHANGE","newValueClass":"long","newValueDisplay":"1","newValuesDictionary":false,"objectClass":"customer","objectId":1,"objectLabel":"DELL","oldValueClass":"long","oldValueDisplay":"0","oldValuesDictionary":false,"property":"salesOpportunitiesCount"}}]}

```

j) Items added to existing documents

POST

<https://system.firmao.pl/{OrganizationID}/svc/v1/transactionentries>

{

"storageDoc":64, // Warehouse Document ID

"transaction":52, // transaction ID

"offer":52, // Bid/order ID

"agreement":52, // Agreement ID

"entryDate":"2024-02-26", **the same date as in the document to which the item belongs**

"warehouse":1, **warehouse ID - in the case of enabled queues, the warehouse must be the same as the document to which the item belongs. If this parameter is not specified, the default warehouse is assigned.**

"mode":"PURCHASE", // **the type of "PURCHASE" or "SALE" must be the same as the document to which the item belongs. If this parameter is not provided, sales are attributed. For warehouse documents, the purchasing type is available for GRN, GRN, GRN.**

"classificationNumber":"","

"discount":null,

"productCode":"DELL001",

"quantity":2,

```

"unit":null,

"vatPercent":23,

"product":50001, product ID, if we provide text (string) instead of a number,
an entry with this text will be created that is not associated with any product

"unitNettoPrice":"100.00"

}

```

The response, in the case of correct creation, also contains data on the change of the document to which the new item has been added

Code 201:

```

{"changelog":[{"deletedWith":"","modification":"ATTACH","newValueClass":"transactionentry","newValueDisplay":{"id":"50081","product":"Dell D630","unit":"","transactionEntryInventory.inventoryQuantity":"","transactionEntryInventory.warehouseQuantity":"","quantity":"2.00","unitNettoPrice":"100.00","unitGrossPrice":"123.00","nettoPrice":"200.00","grossPrice":"246.00","vatPrice":"46.00","vatPercent":"23","productCode":"DELL001","entrySize":"1.00","salePriceGroup":"A","unitPurchaseNettoPrice":"","classificationNumber":"","customsCode":"","profitMarginL":"","margin":"","outcomeType":""},"newValueId":"-1","newValuesDictionary":false,"objectClass":"storagedoc","objectId":64,"objectLabel":"2024/02/015/PZ","oldValuesDictionary":false,"property":"transactionEntries"},{"deletedWith":"","modification":"CREATE","newValuesDictionary":false,"objectClass":"transactionentry","objectId":50081,"oldValuesDictionary":false,"property":""},"{"deletedWith":"","modification":"CHANGE","newValueClass":"float","newValueDisplay":"526.00","newValuesDictionary":false,"objectClass":"storagedoc","objectId":64,"objectLabel":"2024/02/015/PZ","oldValueClass":"float","

```

oldValueDisplay":"280.00","oldValuesDictionary":false,"property":"baseGrossPrice"},{"deletedWith":"","modification":"CHANGE","newValueClass":"float","newValueDisplay":"480.00","newValuesDictionary":false,"objectClass":"storagedoc","objectId":64,"objectLabel":"2024/02/015/PZ","oldValueClass":"float","oldValueDisplay":"280.00","oldValuesDictionary":false,"property":"baseNettoPrice"},{"deletedWith":"","modification":"CHANGE","newValueClass":"float","newValueDisplay":"526.00","newValuesDictionary":false,"objectClass":"storagedoc","objectId":64,"objectLabel":"2024/02/015/PZ","oldValueClass":"float","oldValueDisplay":"280.00","oldValuesDictionary":false,"property":"baseRestOfPaidValue"},{"deletedWith":"","modification":"CHANGE","newValueClass":"float","newValueDisplay":"46.00","newValuesDictionary":false,"objectClass":"storagedoc","objectId":64,"objectLabel":"2024/02/015/PZ","oldValueClass":"float","oldValueDisplay":"0.00","oldValuesDictionary":false,"property":"baseVatPrice"},{"deletedWith":"","modification":"CHANGE","newValueClass":"float","newValueDisplay":"526.00","newValuesDictionary":false,"objectClass":"storagedoc","objectId":64,"objectLabel":"2024/02/015/PZ","oldValueClass":"float","oldValueDisplay":"280.00","oldValuesDictionary":false,"property":"restOfPaidValue"},{"deletedWith":"","modification":"CHANGE","newValueClass":"float","newValueDisplay":"526.00","newValuesDictionary":false,"objectClass":"storagedoc","objectId":64,"objectLabel":"2024/02/015/PZ","oldValueClass":"float","oldValueDisplay":"280.00","oldValuesDictionary":false,"property":"storageDocGrossPrice"},{"deletedWith":"","modification":"CHANGE","newValueClass":"float","newValueDisplay":"480.00","newValuesDictionary":false,"objectClass":"storagedoc","objectId":64,"objectLabel":"2024/02/015/PZ","oldValueClass":"float","oldValueDisplay":"280.00","oldValuesDictionary":false,"property":"storageDocNettoPrice"},{"deletedWith":"","modification":"CHANGE","newValueClass":"float","newValueDisplay":"46.00","newValuesDictionary":false,"objectClass":"storagedoc","objectId":64,"objectLabel":"2024/02/015/PZ","oldValueClass":"float

```

", "oldValueDisplay": "0.00", "oldValuesDictionary": false, "property": "storageDocVatPrice"},
{"deletedWith": "", "modification": "CHANGE", "newValueClass": "float", "newValueDisplay": "
1125.00", "newValuesDictionary": false, "objectClass": "product", "objectId": 50001, "objectL
abel": "Dell D630"
, "oldValueClass": "float", "oldValuesDictionary": false, "property": "netPriceInStore"}, {"delet
edWith": "", "modification": "CHANGE", "newValueClass": "float", "newValueDisplay": "26.00
00000000", "newValuesDictionary": false, "objectClass": "product", "objectId": 50001, "objec
tLabel": "Dell
D630", "oldValueClass": "float", "oldValueDisplay": "24.0000000000", "oldValuesDictionary"
: false, "property": "currentStoreState", "subClass": "warehouse", "subId": 1, "subLabel": "War
ehouse
1"}, {"deletedWith": "", "modification": "CHANGE", "newValueClass": "float", "newValueDispla
y": "1325.00", "newValuesDictionary": false, "objectClass": "product", "objectId": 50001, "obje
ctLabel": "Dell
D630", "oldValueClass": "float", "oldValueDisplay": "1125.00", "oldValuesDictionary": false, "
property": "netPriceInStore", "subClass": "warehouse", "subId": 1, "subLabel": "Warehouse
1"}]}

```

3. Object modification (PUT)

Modification of objects is done by means of a PUT request and should be directed to the address

<https://system.firmao.pl/{OrganizationID}/svc/v1/{xxx}/{id}xxx> - name of the object type

id - object identifier

OrganizationIdentifier - organization identifier visible in the url after logging in to the company

Data type:

ContentType = application/json; charset=UTF-8

All fields available in the objects are described in the data import help.

Custom fields have the keys "customFields.custom5" and take strings as values.

Available fields can also be "previewed" by retrieving the data using the GET method (excluding the permission field, which is enumerated and cannot be provided in the POST/PUT request).

Response codes: The API uses standard HTTP error codes.

200 - correct editing
4xx and 5xx - an error occurred while editing the object.

In general, the API is designed to modify multiple fields in a single request, but it is mainly tested for modifying one field at a time, so unexpected problems may occur if you modify multiple fields in a single request.

a) Change of product name:

Modification of the product name with id 50001:

https://system.firmao.pl/{OrganizationID}/svc/v1/products/50001

PUT

```
{  
  "name":"Example product rename"  
}
```

Answer:

```
{"changelog":[{"deletedWith":"","modification":"CHANGE","newValueClass":"string","newValueDisplay":"Example Product Rename","newValuesDictionary":false,"objectClass":"product","objectId":50001,"objectLabel":"Example Product Rename","oldValueClass":"string","oldValueDisplay":"Example Product","oldValuesDictionary":false,"property":"name"}]}
```

In response, the most important fields are

property - name of the changed field

newValueDisplay - new field value

oldValueDisplay - old field value

b) Change of the contractor's name:

Modification of the name of the contractor with id 50001:

<https://system.firmao.pl/{OrganizationID}/svc/v1/customers/50001>

PUT

```
{  
  
    "name":"KOMP-SERVICE"  
  
}
```

Answer:

```
{"changelog":[{"deletedWith":"","modification":"CHANGE","newValueClass":"string","newValueDisplay":"KOMP-SERVICE","newValueIsDictionary":false,"objectClass":"customer","objectId":50001,"objectLabel":"q","oldValueClass":"string","oldValueDisplay":"q","oldValueIsDictionary":false,"property":"name"}]}
```

c) Changing the product inventory (required to enable the organization's setting: Direct modification of stock levels)

<https://system.firmao.pl/{OrganizationID}/svc/v1/products/50001>

PUT

```
{  
  
  "currentStoreState":50  
  
}
```

The answer (the answer is so big because the server calculates various data itself, in this case it calculates the value of products in the warehouse):

```
{"changelog":[{"deletedWith":"","modification":"CHANGE","newValueClass":"float","newValueDisplay":"50","newValuesDictionary":false,"objectClass":"product","objectId":50001,"objectLabel":"Sample product rename","oldValueClass":"float","oldValueDisplay":"0.000","oldValuesDictionary":false,"property":"currentStoreState","subClass":"warehouse","subId":1,"subLabel":"Warehouse 1"},{"deletedWith":"","modification":"CHANGE","newValueClass":"float","newValueDisplay":"0.00","newValuesDictionary":false,"objectClass":"product","objectId":50001,"objectLabel":"Sample Product Rename","oldValueClass":"float","oldValuesDictionary":false,"property":"netPriceInStore"}]}
```

4. Object Fetching (GET)

The download is done via a GET request and should be directed to

<https://system.firmao.pl/{OrganizationID}/svc/v1/{xxx}xxx> - name of the object type

OrganizationIdentifier - organization identifier visible in the url after logging in to the company

Paging

In GET requests, we are dealing with paging of results. This is done by using the start parameters (from which result we start listing on the page), limit (the maximum number of results on the page). The maximum limit value depends on the license purchased. In the response from the server there is a totalSize field that tells about the total number of objects, so we know if there are other pages of objects.

NOTE: In some cases, a totalSize of -1 appears in the response. This means that there are at least as many results as the limit, which means that there is most likely another page of results.

Time limits:

Daily limit: valid from 7:00 a.m. to 5:00 p.m.

Night Limit: Runs from 17:00 to 06:00 and is quadrupled (4x multiplier).

Score limits per plan:

SUPREME: 20,000

STANDARD: 2,000

ENTERPRISE: 10,000

SIMPLE: 1,000

PROFESSIONAL: 50,000

Performance Notes:

The higher the result limit and the more data the user wants to get, the longer the

loading time can be.

Collation

In the request, we can also specify the field by which we sort (sort parameter) and the direction of sorting (dir parameter). If there are no sorting parameters, the results will be sorted by default

Filtering

In the request, we can specify filtering. The filtering format is:

FieldName(ComparisonType), where the field name is the field by which we filter and the ComparisonType is:

contains - whether the field contains a value (for some fields it supports values after semicolons, e.g. "ala; cat"),

notContains - whether the field does not contain a value (for some fields it supports values after semicolons, e.g. "ala; cat"),

eq - whether the field is equal to the value ,

ne - whether the area is not equal to the value ,

in - whether the field is equal to the values after semicolons, e.g. 1; 3; 5 - Mostly Supported ID

notin - is the field not equal to the values after semicolons, e.g. 1; 3; 5 - Mostly Supported ID

gt - whether the field is larger (only for numeric and date fields),

ngt - whether the field is not greater (i.e. less than or equal to) (only for numeric and

date fields),

ge - whether the field is greater than or equal to (only for numeric and date fields),

nge - if the field is not greater than or equal to (i.e. less) than (only for numeric and date fields),

lt - if the field is less than (only for numeric and date fields)

nl - whether the field is not less than (i.e. greater than or equal to) (only for numeric

fields and dates)le - whether the field is less than or equal to (only for numeric and date fields)

nle - whether the field is not smaller (i.e. larger) than (only for numeric and date fields)

Note: Not all fields of each object support all types of filtering.

Examples of filtering:

[https://system.firmao.pl/{idenfyfikatorOrganizacji }/svc/v1/projects?id\(eq\)=15](https://system.firmao.pl/{idenfyfikatorOrganizacji }/svc/v1/projects?id(eq)=15)

downloads the project with ID 15.

[https://system.firmao.pl/{idenfyfikatorOrganizacji }/svc/v1/projects?id\(ge\)=15](https://system.firmao.pl/{idenfyfikatorOrganizacji }/svc/v1/projects?id(ge)=15)

Retrieves a project with an ID greater than or equal to 15.

[https://system.firmao.pl/{idenfyfikatorOrganizacji }/svc/v1/projects?name\(contains\)=proj](https://system.firmao.pl/{idenfyfikatorOrganizacji }/svc/v1/projects?name(contains)=proj)

Downloads projects with a name containing the string "proj".

Response codes

The API uses standard HTTP error codes.

200 - correct download of 4xx and 5xx data - an error occurred while downloading data.

a) Downloading projects

GET

```
https://system.firmao.pl/{idenfyfikatorOrganizacji  
}/svc/v1/projects?start=0&limit=50&sort=name&dir=ASC&name(contains)=przy&dataFor  
mat=MEDIUM
```

Answer:

```
{  
  "date": [  
    {
```

"actualCost": 0,
"actualCostSummaryGross": 0,
"actualCostSummaryNet": 0,
"actualIncomesGross": 0,
"actualIncomesNet": 0,
"actualIncomeSummaryGross": 0,
"actualIncomeSummaryNet": 0,
"actualOutgoingsGross": 0,
"actualOutgoingsNetto": 0,
"actualProfitSummaryGross": 0,
"actualProfitSummaryNet": 0,
"actualTaskProfit": 0,
"actualTransactionProfitGross": 0,
"actualTransactionProfitNet": 0,
"actualWorkHours": null,
"approved": false,
"attachmentsCount": 0,
"autoCalcEstimatedHours": true,
"autoCalcPlannedCost": true,

```
"autoCalcPlannedIncome": true,  
  
"balance": 0,  
  
"budget": 0,  
  
"contact": null,  
  
"costRealizationPercent": 0,  
  
"createdBy": {  
    "id": 1,  
    "label": "Jacob Smith"  
},  
  
"creationDate": "2017-09-21T13:08:59Z",  
  
"customer": null,  
  
"customFields": null,  
  
"description": "Description",  
  
"endDate": null,  
  
"estimatedHours": null,  
  
"finalProfitSummaryGross": 0,  
  
"finalProfitSummaryNet": 0,  
  
"finalTransactionProfitGross": 0,  
  
"finalTransactionProfitNet": 0,
```

```
"hourlyPrice": null,

"hourlyRate": null,

"id": 50001,

"income": 0,

"incomeRealizationPercent": 0,

"lastUpdateDate": "2017-09-21T13:08:59Z",

"managers": [

    {

        "id": 10,

        "label": "Denise Taylor"

    }

],

"margin": null,

"name": "Example",

"overdraftSummaryGross": 0,

"overdraftSummaryNet": 0,

"permissions": {

    "project": [

        "basic_update",
```

```
"costs",  
  
"create",  
  
"extended_update",  
  
"read",  
  
"update"  
  
],  
  
"task": [  
  
"basic_update",  
  
"costs",  
  
"create",  
  
"read",  
  
"update"  
  
]  
  
},  
  
"plannedCost": 0,  
  
"plannedCostSummaryGross": 0,  
  
"plannedCostSummaryNet": 0,  
  
"plannedIncome": 0,  
  
"plannedIncomesGross": 0,
```

```
"plannedIncomesNet": 0,  
  
"plannedIncomeSummaryGross": 0,  
  
"plannedIncomeSummaryNet": 0,  
  
"plannedOutgoingsGross": 0,  
  
"plannedOutgoingsNet": 0,  
  
"plannedProfitSummaryGross": 0,  
  
"plannedProfitSummaryNet": 0,  
  
"plannedTaskProfit": 0,  
  
"plannedTransactionProfitGross": 0,  
  
"plannedTransactionProfitNet": 0,  
  
"purchaseTransactionGross": 0,  
  
"purchaseTransactionNet": 0,  
  
"restrictedTeamMembers": [],  
  
"saleTransactionGross": 0,  
  
"saleTransactionNet": 0,  
  
"startDate": "2012-07-16T22:00:00Z",  
  
"tags": [  
  
  {  
  
    "color": "#FFFFFF",
```

```
    "id": 3,  
    "label": "Project approved"  
  }  
],  
"teamMembers": [  
  {  
    "id": 6,  
    "label": "Emma Clark"  
  }  
],  
"transactionsBalanceGross": 0,  
"transactionsBalanceNet": 0  
}  
],  
"dir": "ASC",  
"sort": "name",  
"totalSize": 1  
}
```

b) Downloading tasks

GET

[https://system.firmao.pl/{OrganizationID}/svc/v1/tasks?start=0&limit=50&sort=name&dir=ASC&name\(contains\)=task&dataFormat=MEDIUM](https://system.firmao.pl/{OrganizationID}/svc/v1/tasks?start=0&limit=50&sort=name&dir=ASC&name(contains)=task&dataFormat=MEDIUM)

Answer:

```
{  
  "date": [  
    {  
      "actualCost": 0,  
      "actualEndDate": null,  
      "actualStartDate": null,  
      "actualWorkHours": 0,  
      "additionalCosts": 0,  
      "address": null,  
      "attachmentsCount": 0,  
      "autoCalcActualCost": true,
```

```
"autoCalcIncome": true,  
  
"autoCalcPlannedCost": true,  
  
"autoCalcPlannedIncome": true,  
  
"autoStatus": false,  
  
"balance": 0,  
  
"contact": null,  
  
"createdBy": {  
    "id": 1,  
    "label": "Jacob Smith"  
},  
  
"creationDate": "2017-09-21T13:11:51Z",  
  
"customer": null,  
  
"customFields": null,  
  
"description": "Description",  
  
"discount": null,  
  
"estimatedHours": 8,  
  
"hasEndDateReminder": false,  
  
"hasStartDateReminder": false,  
  
"hasSubtasks": false,
```

```
"id": 50033,  
  
"income": 0,  
  
"lastUpdateDate": "2017-09-21T13:11:51Z",  
  
"letter": null,  
  
"name": "Example task",  
  
"orderBruttoPrice": null,  
  
"orderCalculateFromGross": false,  
  
"orderNettoPrice": null,  
  
"orderQuantity": null,  
  
"orderUnit": null,  
  
"orderUnitPrice": null,  
  
"orderVatPercent": null,  
  
"orderVatPrice": null,  
  
"parent": null,  
  
"percentDone": 0,  
  
"permissions": {  
    "task": [  
        "basic_update",  
        "costs",
```

```
        "create",
        "read",
        "update"
    ]
},
"plannedCost": 200,
"plannedEndDate": "2012-07-17T22:00:00Z",
"plannedEndDateType": "EXACTLY",
"plannedIncome": 400,
"plannedStartDate": "2012-07-16T22:00:00Z",
"plannedStartDateType": "EXACTLY",
"priority": 300,
"product": null,
"project": null,
"reminderEndDateAdvance": null,
"reminderStartDateAdvance": null,
"resource": null,
"responsibleUsers": [
    {
```

```
    "id": 2,  
    "label": "Ethan Lam"  
  }  
],  
"status": {  
  "group": "OPEN",  
  "icon": null,  
  "key": "WAITING",  
  "label": "Pending",  
  "label_de": "Wird erwartet",  
  "label_en": "Waiting"  
},  
"subcontact": null,  
"subcontractor": null,  
"subtasksCount": 0,  
"Tags": [],  
"taskType": "PROJECT",  
"transactionGrossPrice": 0,  
"transactionNettoPrice": 0
```

```
}  
  
],  
  
"dir": "ASC",  
  
"sort": "name",  
  
"totalSize": 1  
  
}
```

c) Downloading products

GET

[https://system.firmao.pl/{identyfikatorOrganizacji \(Organization Identifier\)}/svc/v1/products?start=0&limit=50&sort=name&dir=ASC&dataFormat=MEDIUM](https://system.firmao.pl/{identyfikatorOrganizacji (Organization Identifier)}/svc/v1/products?start=0&limit=50&sort=name&dir=ASC&dataFormat=MEDIUM)

Answer:

```
{  
  
  "date": [  
  
    {  
  
      "classificationNumber": null,
```

```
"createdBy":{  
  
  "id":1,  
  
  "label":"Jacob Smith"  
  
},  
  
"creationDate":"2012-07-17T08:39:15Z",  
  
"currentStoreState":50.00,  
  
"customFields":null,  
  
"description":null,  
  
"endSellingDate":null,  
  
"grossPrice":12.30,  
  
"grossPriceInStore":615.00,  
  
"id":50001,  
  
"lastUpdateDate":"2012-07-17T08:39:15Z",  
  
"margin":20.00,  
  
"name":"Example product rename",  
  
"netPrice":10.00,  
  
"netPriceInStore":500.00,  
  
"overrunState":true,  
  
"permissions":{
```

```
"product":[
    "buyprice",
    "create",
    "extended_update",
    "read",
    "update"
],
"semiproduct":[
    "create",
    "read",
    "update"
],
"transactionentry":[
    "create",
    "read",
    "update"
]
},
"productCode":"PR1",
```

```
    "saleGrossPrice":14.76,  
  
    "saleGrossPriceInStore":738.00,  
  
    "saleNetPrice":12.00,  
  
    "saleNetPriceInStore":600.00,  
  
    "saleable":true,  
  
    "startSellingDate":null,  
  
    "tags":[  
  
    ],  
  
    "unit":"pcs",  
  
    "vat":23.00,  
  
    "warningStoreState":88.00  
  
    }  
  
],  
  
"dir":"ASC",  
  
"sort":"name",  
  
"totalSize":1  
  
}
```

d) Downloading customers

GET

<https://system.firmao.pl/{idenfyikatorOrganizacji}/svc/v1/customers?start=0&limit=50&sort=name&dir=ASC&dataFormat=MEDIUM>

Answer:

```
{  
  
  "date": [  
  
    {  
  
      "actualCost": 0.00,  
  
      "balance": 0.00,  
  
      "bankAccountNumber": "50 1020 5558 1111 1702 0600 0067",  
  
      "correspondenceAddress": null,  
  
      "createdBy": null,  
  
      "creationDate": "2002-12-13T11:40:45Z",  
  
      "customFields": null,  
  
      "customerGroups": [  

```

```
],  
  
"customerType":"PARTNER",  
  
"description":"A company specializing in computer service",  
  
"email":"kompservice.firmao@mailinator.com",  
  
"email2":"komputery@mailinator.com",  
  
"email3":null,  
  
"emails":[  
  
    "kompservice.firmao@mailinator.com",  
  
    "komputery@mailinator.com"  
  
],  
  
"employeesNumber":1,  
  
"faxNumber":"22 398 78 87",  
  
"hourlyRate":null,  
  
"id":51,  
  
"identificationNumber":null,  
  
"income":0.00,  
  
"industry":{  
  
    "key":"3",  
  
    "label":"IT"
```

```
},  
  
"krs":null,  
  
"label":"KOMP-SERVICE",  
  
"lastSalesNote":null,  
  
"lastUpdateDate":"2010-05-29T20:53:30Z",  
  
"name":"KOMP-SERVICE Jan Kowalski",  
  
"nipNumber":"7291077843",  
  
"officeAddress":{  
  
  "city":"Pruszków",  
  
  "country":"Poland",  
  
  "county":null,  
  
  "postCode":"05-800",  
  
  "street":"Stefana Batorego 3/4"  
  
},  
  
"ownership":"PRIVATE",  
  
"permissions":{  
  
  "vatrecord":[  
  
    "create",  
  
    "read",
```

```
        "update"
    ],
    "task":[
        "basic_update",
        "costs",
        "create",
        "read",
        "update"
    ],
    "accountingoperation":[
        "create",
        "read",
        "update"
    ],
    "customer":[
        "costs",
        "create",
        "read",
        "update"
```

```
]
},
"phone":"22 345 74 26",
"phoneOther":null,
"phoneOther2":null,
"phones":[
  "22 345 74 26"
],
"plannedCost":0.00,
"plannedIncome":0.00,
"stats":null,
"status":null,
"tags":[

],
'vatUE':'EN 7291077843',
"voivodeship":null,
"website": "www.kompservice.bis"
}
```

```
],  
  
"dir":"DESC",  
  
"sort":"id",  
  
"totalSize":1  
  
}
```

e) Downloading transactions

GET

<https://system.firmao.pl/{identyfikatorOrganizacji}/svc/v1/transactions?start=0&limit=50&sort=name&dir=ASC&dataFormat=MEDIUM>

Answer:

```
{  
  
"date":[  
  
  {  
  
    "actualPaymentDate":null,  
  
    "actualPaymentType":null,  
  
    "advanceInvoiceGroupId":null,
```

```
"approved":false,

"calculateFromGross":false,

"createdBy":null,

"creationDate":"2002-12-13T11:40:45Z",

"currency":"PLN",

"customFields":null,

"customer":{

  "emails":[

    "icenter.firmao@mailinator.com",

    "iceneter.biuro1@mailinator.com",

    "iceneter.biuro2@mailinator.com"

  ],

  "id":50,

  "label":"Internet Center"

},

"customerAddress":null,

"customerName":null,

"description":null,

"hasReminder":false,
```

"id":50,
"invalidInvoiceDate":null,
"invalidTransactionNumber":null,
"invoiceAnnotations":null,
"invoiceDate":"2010-09-27",
"issuingPerson":null,
"lastUpdateDate":"2010-05-29T20:53:30Z",
"mail":null,
"mode":"SALE",
"name":"Sale of DE PC",
"nipNumber":null,
"paid":true,
"paidValue":999,
"parcelCourierName":null,
"parcelDeliveryType":null,
"parcelDescription":null,
"parcelNumber":null,
"parcelShipmentDate":null,
"paymentDate":"2010-09-30",

```
"paymentStatus":null,
"paymentType":"TRANSFER",
"permissions":{
  "transaction":[
    "create",
    "extended_update",
    "read",
    "update"
  ],
  "transactionentry":[
    "create",
    "read",
    "update"
  ]
},
"project":{
  "id":4,
  "name":"GANT USE CASE2"
},
```

```
"reminderAdvance":null,

"restOfPaidValue":0,

"sellerAddress":null,

"storageStateUpdate":"NO_UPDATE",

"tags":[

],

"task":{

  "id":24,

  "name":"GANTT UC2 C",

  "status":{

    "group":"OPEN"

  }

},

"transactionGrossPrice":999,

"transactionDate":"2010-09-24",

"transactionGroupBruttoPrice":null,

"transactionGroupVatPrice":null,

"transactionNettoPrice":818.85,
```

```
    "transactionNumber":"1/1/2010",  
    "transactionVatPrice":180.15,  
    "type":"INVOICE"  
  }  
],  
  "dir":"ASC",  
  "sort":"name",  
  "totalSize":1  
}
```

f) Downloading transactions for a counterparty

GET

[https://system.firmao.pl/{OrganizationID}/svc/v1/transactions?start=0&limit=50&customer\(eq\)=10](https://system.firmao.pl/{OrganizationID}/svc/v1/transactions?start=0&limit=50&customer(eq)=10)

where 10 is the contractor's ID

Answer:

```
{  
  "date": [  
    {  
      "actualPaymentDate": null,  
      "actualPaymentType": null,  
      "advanceInvoiceGroupId": null,  
      "approved": true,  
      "attachmentsCount": 0,  
      "bankAccount": {  
        "id": 1,  
        "label": "test"  
      },  
      "baseBruttoPrice": 0,  
      "baseNettoPrice": 0,  
      "basePaidValue": 0,  
      "baseRestOfPaidValue": 0,  
      "baseVatPrice": 0,  
      "calculateFromGross": false,  
      "commision": 0,  
    },  
  ],  
}
```

```
"commisionRate": 0,  
  
"connectedDocDate": null,  
  
"connectedDocNumber": null,  
  
"connectedDocType": null,  
  
"contact": null,  
  
"correctionCause": null,  
  
"createdBy": {  
  "id": 1,  
  "label": "Jacob Smith"  
},  
  
"creationDate": "2017-09-21T13:38:08Z",  
  
"currency": "PLN",  
  
"customer": {  
  "discount": null,  
  "emails": null,  
  "id": 1,  
  "label": "DELL",  
  "salePriceGroup": "A"  
},
```

"customerAddress": null,
"customerBankAccountNumber": null,
"customerName": "DELL",
"customFields": null,
"dateForExchangeRate": "2017-09-21",
"dateLabel": "SELL_DATE",
"description": null,
"documentDate": "2017-09-21",
"documentGenerated": false,
"documentNumber": "2017/9/21/21",
"doNotIncludeInLedgerReport": false,
"exchangeRate": null,
"exchangeRateDate": null,
"factor": 1,
"finalPartialInvoice": false,
"hasReminder": false,
"id": 50034,
"integrationType": null,
"invoiceAnnotations": null,

```
"invoiceDate": "2017-09-21",  
  
"invoicePatternId": 1,  
  
"invoiceReceptionDate": null,  
  
"issuingPerson": "Jacob Smith",  
  
"issuingPersonObject": {  
    "firstName": "Jacob",  
  
    "id": 1,  
  
    "lastName": "Smith"  
},  
  
"lastSelectedInvoiceReport": "Invoice",  
  
"lastUpdateDate": "2017-09-21T13:38:08Z",  
  
"mail": null,  
  
"mode": "SALE",  
  
"name": null,  
  
"nipNumber": null,  
  
"paid": false,  
  
"paidValue": 0,  
  
"paidValueInvoice": 0,  
  
"parcelCourierName": null,
```

```
"parcelDeliveryType": null,

"parcelDescription": null,

"parcelNumber": null,

"parcelShipmentDate": null,

"paymentDate": "2017-09-21",

"paymentStatus": "NOT_PAID",

"paymentType": "CASH",

"permissions": {

  "transaction": [

    "create",

    "extended_update",

    "read",

    "update"

  ]

},

"phone": "+48 42 356 53 34",

"profit": 0,

"profitCommision": 0,

"profitCommisionRate": 0,
```

"projects": [],
"purchaseVatCategory": null,
"rateFromCurrency": null,
"recipient": null,
"recipientAddress": null,
"recipientName": null,
"recipientNipNumber": null,
"recipientPhoneNumber": null,
"reminderAdvance": null,
"restOfPaidValue": 0,
"restOfPaidValueInvoice": 0,
"sellerAddress": {},
"sellerBankAccountNumber": "PL83 1010 1023 0000 2613 9510 0000",
"smallTaxPayer": false,
"storageStateUpdate": "NO_UPDATE",
"Tags": [],
"task": null,
"transactionGrossPrice": 0,
"transactionDate": "2017-09-21",

```
"transactionGroupGrossPrice": 0,  
  
"transactionGroupVatPrice": 0,  
  
"transactionNettoPrice": 0,  
  
"transactionNettoPriceTax": 0,  
  
"transactionNumber": "2017/9/21/21",  
  
"transactionOrInvoiceDate": "2017-09-21",  
  
"transactionVatPeriod": "09.2017",  
  
"transactionVatPrice": 0,  
  
"type": "INVOICE",  
  
"warehouse": {  
  "id": 1,  
  "name": "Warehouse 1"  
}  
  
},  
  
"dir": "DESC",  
  
"sort": "transactionDate",  
  
"totalSize": 1  
  
}
```

g) retrieving transaction items - transactions

GET

[https://system.firmao.pl/{identyfikatorOrganizacji}/svc/v1/transactionentries?transaction\(eq\)=50](https://system.firmao.pl/{identyfikatorOrganizacji}/svc/v1/transactionentries?transaction(eq)=50)

where 50 is the transaction id

Answer:

```
{  
  "date": [  
    {  
      "balance": null,  
      "grossPrice": 999.00,  
      "calculateFromGross": false,  
      "classificationNumber": null,  
      "description": null,  
    }  
  ]  
}
```

```
"discount":null,

"entryDate":"2010-09-24",

"id":50,

"invalidEntry":null,

"mode":"SALE",

"name":null,

"nettoPrice":818.85,

"offer":null,

"permissions":{

  "transactionentry":[

    "create",

    "read",

    "update"

  ]

},

"product":{

  "id":51,

  "name":"De PC Pentium 2.74 2GB"

},
```

```
"productCode":"ST0002",  
  
"project":null,  
  
"quantity":1.000,  
  
"transaction":{  
  "customerName":null,  
  
  "id":50,  
  
  "name":"Sale of DE PC",  
  
  "transactionNumber":"1/1/2010"  
},  
  
"type":"ACTUAL",  
  
"unit":"pcs",  
  
"unitBruttoPrice":999.00,  
  
"unitNettoPrice":818.85,  
  
"vatPercent":22.00,  
  
"vatPrice":180.15  
}  
  
],  
  
"dir":"ASC",  
  
"sort":"transaction",
```

```
"totalSize":1  
}
```

h) downloading offers/orders

GET

<https://system.firmao.pl/{idenfyfikatorOrganizacji}/svc/v1/offers?start=0&limit=50&sort=name&dir=ASC&dataFormat=MEDIUM>

```
{  
  "date":[  
    {  
      "annotations":null,  
      "grossPrice":999,  
      "calculateFromGross":false,  
      "contact":null,  
      "createdBy":{  
        "id":1,  
        "label":"Jacob Smith"      }  
    }  
  ]  
}
```

```
},  
  
"creationDate":"2012-11-09T12:06:47Z",  
  
"currency":"PLN",  
  
"customFields":null,  
  
"customer":{  
  
  "email":"icenter.firmao@mailinator.com",  
  
  "id":50,  
  
  "label":"Internet Center"  
  
},  
  
"customerAddress":{  
  
  "city":"Łódź",  
  
  "country":"Poland",  
  
  "county":null,  
  
  "postCode":"91-477",  
  
  "street":"Piotrkowska 147/8"  
  
},  
  
"customerName":"Internet Center dsfds",  
  
"daysToDueDate":7,  
  
"description":null,
```

"hasReminder":false,
"id":1,
"issuingPerson":"Jacob Smith",
"lastUpdateDate":"2012-11-09T12:06:48Z",
"mail":null,
"mode":"SALE",
"name":null,
"nettoPrice":818.85,
"nipNumber":"1060000062",
"number":"1/11/2012/U",
"offerDate":"2012-11-09",
"offerStatus":"NEW",
"parcelCourierName":null,
"parcelDeliveryType":null,
"parcelDescription":null,
"parcelNumber":null,
"parcelShipmentDate":null,
"paymentDate":"2012-11-09",
"paymentType":"CASH",

```
"permissions":{  
  "offer":[  
    "create",  
    "read",  
    "update"  
  ],  
  "transactionentry":[  
    "create",  
    "read",  
    "update"  
  ]  
},  
"project":null,  
"reminderAdvance":null,  
"sellerAddress":{  
  
},  
"tags":[
```

```
    ],  
    "task":null,  
    "type":"AGREEMENT",  
    "validFrom":"2012-11-09",  
    "vatPrice":180.15  
  }  
],  
"dir":"DESC",  
"sort":"offerDate",  
"totalSize":1  
}
```

i) download of sales opportunities

GET

<https://system.firmao.pl/{identyfikatorOrganizacji}/svc/v1/salesopportunities?start=0&limit=50&sort=name&dir=ASC&dataFormat=MEDIU>

M

```
{
```

```
"date":{
  "acquisitionMethod":null,
"attachmentsCount":0,
"calculateFromGross":false,
"checklistEntrySelections":[],
"contact":null,
"createdBy":{
  "id":1,
  "label":"Jacob Smith"},
  "creationDate":"2023-09-11T05:28:14Z",
  "currency":"PLN",
  "customFields":null,
  "customer":{
    "emails":["dell.firmao@mailinator.com"],
    "id":1,
    "label":"DELL",
    "paymentDate":null,"phones":["+48 42 356 53
34"],"voivodeship":null},"description":null,"hasReminder":false,"id":8,"label":"test","lastUp
dateDate":"2023-09-11T05:28:13Z","mailsCount":0,"nextContactDate":null,"observers":[]
,"offersCount":0,"openTaskDate":null,"outdatedTaskCount":0,"permissions":{"salesoppor
```

```

tunity":["comment","create","observer_update","read","update"]},"phoneCallsCount":0,"p
roductsCount":0,"reminderAdvance":null,"remindersCount":0,"resignationCause":null,"re
sponsibleUser":{"id":1,"label":"Jacob
Smith"},"salesDate":null,"salesNotesCount":0,"salesOpportunityValue":0.00,"salesOppor
tunityValueInPLN":0.00,"sellingProcess":{"color":"#FFFFFF","key":"DEFAULT","label":"D
efault selling process","label_ar":"عملية البيع
الافتراضية","label_de":"Standardverkaufsprozess","label_en":"Default selling
process","label_es":"Proceso de venta por defecto","label_fr":"Processus de vente par
défaut","label_hr":"Taskí proces prodaje","label_jp":"既定の販売プロセス
","label_pt":"Processo de venda por defeito","label_ru":"Процесс продажи по
умолчанию","label_se":"Standardförsäljningsprocess","label_tr":"Varsayılan satış
süreci","label_ua":"Воронка продажу за
замовчуванням"},"stage":{"color":"#B94747","key":"3","label":"Lead
acquired","label_ar":"تم الحصول على الرصاص","label_de":"Blei erworben","label_en":"Lead
acquired","label_es":"Plomo adquirido","label_fr":"Piste acquise","label_hr":"Stečeno
vodstvo","label_jp":"獲得した見込客","label_pt":"Chumbo adquirido","label_ru":"Клиент
приобрел","label_se":"Bly som förvärvats","label_tr":"Kurşun elde
edildi","label_ua":"Клієнт
здобутий"},"status":"OPEN","statusChangeDate":null,"tags":[],"taskForTodayCount":0,"t
askToShowOnKanban":null,"tasksCount":2,"transactionsCount":0}],

"dir":"DESC",

"sort":"creationDate",

"totalSize":1

}

```

j) Getting contacts

GET

<https://system.firmao.pl/{idenfyfikatorOrganizacji}/svc/v1/contacts?start=0&limit=50&sort=firstName&dir=ASC&dataFormat=MEDIUM>

```
{  
  
  "date": [  
  
    {  
  
      "address": null,  
  
      "createdBy": null,  
  
      "creationDate": "2002-12-13T11:40:45Z",  
  
      "customer": {  
  
        "emails": [  
  
          "iceneter.biuro1@mailinator.com",  
  
          "iceneter.biuro2@mailinator.com",  
  
          "icenter.firmao@mailinator.com"  
  
        ],  
  
        "id": 50,  
  
        "label": "Internet Center"  
  
      },  
  
    },  
  
  ],  
  
}
```

```
"customFields": null,

'Department': 'Secretariat',

"description": null,

"email": "anna.wojewodzka@mailinator.com",

"email2": null,

"email3": null,

"emails": [

    "anna.wojewodzka@mailinator.com"

],

"firstName": "Anna",

"hasReminder": false,

"id": 51,

"instantMessenger": null,

"label": "Anna Wojewódzka",

"lastName": "Wojewódzka",

"lastUpdateDate": "2010-05-29T20:53:30Z",

"nextContactDate": null,

"permissions": {

    "contact": [
```

```
        "create",
        "read",
        "update"
    ]
},
"phone": "42 688 14 22",
"phoneOther": null,
"phoneOther2": null,
"phones": [
    "42 688 14 22"
],
"position": "Secretary",
"reminderAdvance": null,
"skypeId": null,
"Tags": []
},
{
"address": null,
"createdBy": null,
```

```
"creationDate": "2002-12-13T11:40:45Z",
"customer": {
  "emails": [
    "iceneter.biuro1@mailinator.com",
    "iceneter.biuro2@mailinator.com",
    "icenter.firmao@mailinator.com"
  ],
  "id": 50,
  "label": "Internet Center"
},
"customFields": null,
"department": null,
"description": null,
"email": "m.nowak@mailinator.com",
"email2": null,
"email3": null,
"emails": [
  "m.nowak@mailinator.com"
],
```

```
"firstName": "Maciej",  
  
"hasReminder": false,  
  
"id": 50,  
  
"instantMessenger": null,  
  
"label": "Maciej Nowak",  
  
"lastName": "Smith",  
  
"lastUpdateDate": "2017-09-21T14:44:57Z",  
  
"nextContactDate": null,  
  
"permissions": {  
    "contact": [  
        "create",  
        "read",  
        "update"  
    ]  
},  
  
"phone": "48507306295",  
  
"phoneOther": "641044884",  
  
"phoneOther2": null,  
  
"phones": [
```

```
        "48507306295",
        "641044884"
    ],
    "position": "Salesperson",
    "reminderAdvance": null,
    "skypeId": null,
    "Tags": []
}
],
"dir": "ASC",
"sort": "firstName",
"totalSize": 2
}
```

k) downloading customergroups

<https://system.firmao.pl/{idenfyfikatorOrganizacji}/svc/v1/customergroups?start=0&limit=50&sort=firstName&dir=ASC&dataFormat=MEDIUM>

```
{
  "date": [
    {
      "createdBy": {
        "id": 2,
        "label": "Ethan Lam"
      },
      "creationDate": "2010-07-13T07:00:00Z",
      "hasTransactions": [],
      "id": 1,
      "lastUpdateDate": "2010-07-13T07:00:00Z",
      "name": "Regular customers",
      "permissions": {
        "customergroup": [
          "create",
          "read",
          "update"
        ]
      }
    ]
  }
}
```

```
},  
  
  "Salesman": [  
  
    {  
  
      "id": 5,  
  
      "label": "Anthony Wilson"  
  
    }  
  
  ]  
  
}  
  
],  
  
"dir": "ASC",  
  
"multiloggingdetected": true,  
  
"sort": "name",  
  
"totalSize": 1  
  
}
```

l) retrieving sales notes - salesnotes

[https://system.firmao.pl/{OrganizationID}/svc/v1/salesnotes?start=0&limit=50&sort=first
Name&dir=ASC&dataFormat=MEDIUM](https://system.firmao.pl/{OrganizationID}/svc/v1/salesnotes?start=0&limit=50&sort=firstName&dir=ASC&dataFormat=MEDIUM)

```
{
```

```
"date": [  
  
  {  
  
    "agreement": null,  
  
    "attachmentsCount": 0,  
  
    "contact": null,  
  
    "createdBy": {  
  
      "id": 1,  
  
      "label": "Jacob Smith"  
  
    },  
  
    "creationDate": "2017-09-12T12:06:19Z",  
  
    "customer": null,  
  
    "customFields": null,  
  
    "date": "2017-09-12T12:06:00Z",  
  
    "description": "meeting",  
  
    "firstForCustomer": false,  
  
    "id": 50006,  
  
    "lastUpdateDate": "2017-09-12T12:06:18Z",  
  
    "mail": null,  
  
    "offer": null,  
  
  },  
  
]
```

```
"permissions": {  
  "salesnote": [  
    "create",  
    "read",  
    "update"  
  ]  
},  
"product": null,  
"project": null,  
"Tags": [],  
"task": null,  
"transaction": null,  
"type": "MEETING"  
}  
],  
"dir": "ASC",  
"multiloggingdetected": true,  
"sort": "date",  
"totalSize": 1
```

```
}
```

m) Downloading warehouse documents - StorageDocs

<https://system.firmao.pl/{idenfyfikatorOrganizacji>

[/svc/v1/storagedocs?start=0&limit=50&sort=firstName&dir=ASC&dataFormat=MEDIUM](https://system.firmao.pl/{idenfyfikatorOrganizacji}/svc/v1/storagedocs?start=0&limit=50&sort=firstName&dir=ASC&dataFormat=MEDIUM)

```
{
```

```
  "date": [
```

```
    {
```

```
      "attachmentsCount": 0,
```

```
      "bankAccount": null,
```

```
      "baseGrossPrice": 29,
```

```
      "baseNettoPrice": 23.58,
```

```
      "basePaidValue": 0,
```

```
      "baseRestOfPaidValue": 29,
```

```
      "baseVatPrice": 5.42,
```

```
      "calculateFromGross": false,
```

```
      "contact": null,
```

```
      "createdBy": {
```

```
        "id": 1,
```

```
    "label": "Jacob Smith"
  },
  "creationDate": "2017-09-13T08:26:34Z",
  "currency": "PLN",
  "customer": {
    "emails": [
      "kompservice.firmao@mailinator.com"
    ],
    "id": 51,
    "label": "KOMP-SERVICE"
  },
  "customerAddress": {
    "city": "Pruszków",
    "country": "Poland",
    "county": null,
    "postCode": "05-800",
    "street": "Stefana Batorego 4"
  },
  "customerBankAccountNumber": "50 1020 5558 1111 1702 0600 0067",
```

```
"customerName": "KOMP-SERVICE Jan Kowalski",  
  
"customFields": null,  
  
"dateForExchangeRate": "2017-09-13",  
  
"description": null,  
  
"documentDate": "2017-09-13",  
  
"documentNumber": "2017/09/001/PZ",  
  
"exchangeRate": null,  
  
"exchangeRateDate": null,  
  
"factor": 1,  
  
"hasReminder": false,  
  
"id": 1,  
  
"invoiceAnnotations": null,  
  
"invoiceDate": "2017-09-13",  
  
"issuingPerson": "Jacob Smith",  
  
"issuingPersonObject": {  
    "firstName": "Jacob",  
  
    "id": 1,  
  
    "lastName": "Smith"  
},
```

```
"lastSelectedInvoiceReport": "Przyjecie_zewnetrzne",  
  
"lastUpdateDate": "2017-09-13T08:26:33Z",  
  
"name": null,  
  
"nipNumber": "7291077843",  
  
"paid": false,  
  
"paidValue": 0,  
  
"parcelCourierName": null,  
  
"parcelDeliveryType": null,  
  
"parcelDescription": null,  
  
"parcelNumber": null,  
  
"parcelShipmentDate": null,  
  
"paymentDate": "2017-09-13",  
  
"paymentType": "CASH",  
  
"permissions": {  
  "storagedoc": [  
    "create",  
    "read",  
    "update"  
  ]  
}
```

```
},  
  
"phone": "22 345 74 26",  
  
"rateFromCurrency": null,  
  
"receivingPerson": null,  
  
"receivingPersonObject": null,  
  
"recipient": null,  
  
"recipientAddress": null,  
  
"recipientName": null,  
  
"recipientNipNumber": null,  
  
"recipientPhoneNumber": null,  
  
"reminderAdvance": null,  
  
"restOfPaidValue": 29,  
  
"sellerAddress": {},  
  
"sellerBankAccountNumber": null,  
  
"sourceWarehouse": null,  
  
"storageDocGrossPrice": 29,  
  
"storagedocDate": "2017-09-13",  
  
"storageDocNettoPrice": 23.58,  
  
"storageDocNumber": "2017/09/001/PZ",
```

```
"storageDocVatPrice": 5.42,  
  
"storageStateUpdate": "AUTO_UPDATE",  
  
"Tags": [],  
  
"task": null,  
  
"type": "OUTSIDE_INCOME",  
  
"warehouse": {  
  "id": 1,  
  "name": "Warehouse 1"  
}  
  
}  
  
],  
  
"dir": "ASC",  
  
"multiloggingdetected": true,  
  
"sort": "name",  
  
"totalSize": 1  
  
}
```

n) downloading the list of files - documents

<https://system.firmao.pl/{identyfikatorOrganizacji}/svc/v1/documents?start=0&limit=50&dataFormat=MEDIUM>

```
{  
  "date": [  
    {  
      "contentType": "image/png",  
      "createdBy": null,  
      "creationDate": "2017-09-08T07:27:05Z",  
      "description": null,  
      "fileSize": 15861,  
      "isFile": true,  
      "name": "specjalista.png",  
      "objectClass": "mail",  
      "objectDeleted": false,  
      "objectId": 23,  
      "objectLabel": "Training Calendar - September 2017",  
      "Tags": [],  
      "uploaded": true,  
    }  
  ]  
}
```

```
"url": null

},

{

  "contentType": "text/plain",

  "createdBy": null,

  "creationDate": "2017-09-08T07:27:08Z",

  "description": null,

  "fileSize": 117,

  "isFile": true,

  "name": "Raport_dostarczenia.txt",

  "objectClass": "mail",

  "objectDeleted": false,

  "objectId": 27,

  "objectLabel": "Mail delivery failed: returning message to sender",

  "Tags": [],

  "uploaded": true,

  "url": null

}

],
```

```
"dir": "ASC",  
  
"sort": "name",  
  
"totalSize": 32  
  
}
```

o) Downloading e-mails - mails

[https://system.firmao.pl/{OrganizationID}/svc/v1/mails?start=0&limit=50&sort=sentDate&dir=DESC&box\(eq\)=INBOX&dataFormat=MEDIUM](https://system.firmao.pl/{OrganizationID}/svc/v1/mails?start=0&limit=50&sort=sentDate&dir=DESC&box(eq)=INBOX&dataFormat=MEDIUM)

```
{  
  
  "changelog": [],  
  
  "date": [  
  
    {  
  
      "answerFor": null,  
  
      "attachment": false,  
  
      "bcc": [],  
  
      "box": "INBOX",  
  
      "cc": [  
  
        "mailadress@mailinator.com"  
      ]  
  
    }  
  
  ]  
  
}
```

```
],  
  
  "contact": null,  
  
  "createdBy": null,  
  
  "customer": {  
    "email": "dell@gmail.com",  
  
    "id": 1,  
  
    "label": "DELL"  
  },  
  
  "customFields": null,  
  
  "done": false,  
  
  "folder": null,  
  
  "id": 58,  
  
  "mailAccount": {  
    "id": 1,  
  
    "incomingServerProtocol": "IMAP",  
  
    "label": "test@firmao.pl"  
  },  
  
  "mailInvitations": [],  
  
  "mailSendingTime": null,
```

```
"project": null,

"read": true,

"replyTo": "\"dell@gmail.com\" <dell@gmail.com>",

"sender": "dell@gmail.com",

"senderLabel": "dell@gmail.com",

"sentDate": "2017-09-08T07:56:50Z",

"status": "RECEIVE_SUCCEED",

"subject": "as",

"Tags": [],

"task": null,

"to": [

    "test@firmao.pl"

]

},

{

"answerFor": null,

"attachment": false,

"bcc": [],

"box": "INBOX",
```

```
"cc": [],  
  
"contact": null,  
  
"createdBy": null,  
  
"customer": {  
  "email": "dell@wp.pl",  
  "id": 50002,  
  "label": "DELL"  
},  
  
"customFields": null,  
  
"done": false,  
  
"folder": null,  
  
"id": 24,  
  
"mailAccount": {  
  "id": 2,  
  "incomingServerProtocol": "IMAP",  
  "label": "dell@gmail.com"  
},  
  
"mailInvitations": [],  
  
"mailSendingTime": null,
```

```
"project": null,

"read": true,

"replyTo": "\"DELL\" <dell@wp.pl>",

"sender": "dell@wp.pl",

"senderLabel": "DELL",

"sentDate": "2017-09-08T07:26:27Z",

"status": "REPLIED",

"subject": "Test Firmao",

"Tags": [],

"task": null,

"to": [

    "mailadress@firmao.pl",

    "dell@gmail.com"

] },

"dir": "DESC",

"sort": "sentDate",

"totalSize": 51

}
```

p) downloading a series of numbering invoices

GET: <https://system.firmao.pl/{OrganizationID}/svc/v1/invoicenumbers?dataFormat=MEDIUM>

The query will retrieve a list of numbering series - their id numbers, names and numbering templates for each type of document (regular invoice / proforma invoice / offer / order, etc.).

The id retrieved in this way can be used, for example, to retrieve the next invoice number:

GET:

<https://system.firmao.pl/{OrganizationID}/svc/v1/invoicenumbers/next?objectClass={ObjectName}&type={DocumentType}&mode={SalesMode}&id={NumberSeriesIDs}>

id: obtained from the previous query

mode:

- SALE - sales document

- PURCHASE - purchase document

objectClass and the type dictionary:

- transaction
 - invoice
 - receipt (Paragon)
 - pro forma invoice
 - correction
 - bill
 - invoicemargin
 - advance_invoice (Advance Invoice)
- agreement
 - agreement
- offer
 - offer
 - order
- storagedoc
 - outside_income (Stock receipt)
 - outside_release (External edition),

- inside_release (Internal issue),
- inside_income (Internal revenue),
- warehouse_transfer (Warehouse Transfer)

For example, in order to retrieve the sequential number of the proforma sales invoice for the stellis organization for the numbering line id 2, the following inquiry should be sent:

GET:

<https://system.firmao.pl/stellis/svc/v1/invoicenumbers/next?objectClass=transaction&type=proforma&mode=SALE&id=2>

For the same numbering line and organization, picking the sequential sales order number is:

GET:

<https://system.firmao.pl/stellis/svc/v1/invoicenumbers/next?objectClass=offer&type=order&mode=SALE&id=2>

To retrieve the next contract number:

GET:

<https://system.firmao.pl/stellis/svc/v1/invoicenumbers/next?objectClass=agreement&type=agreement&mode=SALE&id=2>

5. Deleting objects (PUT, DELETE)

In firmao, most objects are not deleted, but marked as deleted and can be restored. Such objects are deleted by modifying the deleted field to true as described in Chapter 3.

Other objects (which do not have the deleted field) are deleted by the DELETE method. These include:- reminders

-Attachments

- transaction items These objects cannot be restored.

6. A detailed description of creating, editing, finishing, and starting work times

Creating a new working time **with only the start time:**

https://system.firmao.pl/{OrganizationID}/svc/v1/timeentries

POST

```
{  
  
  "type":"TASK",  
  
  "user":1,  
  
  "dateTimeFrom":"2017-09-15T08:30:00+01:00"  
  
}
```

Answer:

```
{ "changelog" :  
  [{"deletedWith":"","modification":"ATTACH","newValueClass":"timeentry","newValueDisplay":"Jacob Smith, CTE\nod  
{[2017-09-15T07:30:00.000Z]}\n","newValueId":"50007","newValuesDictionary":false,"  
objectClass":"user","objectId":1,"objectLabel":"Jacob  
Smith","oldValuesDictionary":false,"property":"timeEntry"}]  
  
}
```

In response , **"newValueId":"50007"** is the ID of the newly created working time

Create a new working time **with only the end time:**

`https://system.firmao.pl/{OrganizationID}/svc/v1/timeentries`

POST

```
{  
  
  "type":"TASK",  
  
  "user":1,  
  
  "dateTimeTo":"2017-09-15T16:30:00+01:00"  
  
}
```

Answer:

```
{ "changelog" :  
  [{"deletedWith":"","modification":"ATTACH","newValueClass":"timeentry","newValueDisplay":"Jacob Smith,  
CTE\ndo{[2017-09-15T16:30:00.000Z]}\n","newValued":"50008","newValuesDictionary":false,"objectClass":"user","objectId":1,"objectLabel":"Jacob  
Smith","oldValuesDictionary":false,"property":"timeEntry"}]  
  
}
```

In response , **"newValued":"50008"** is the ID of the newly created working time

End of **existing** working time

<https://system.firmao.pl/{idenfyfikatorOrganizacji }/svc/v1/timeentries/50007>

PUT

```
{
    "dateTimeTo":"2017-09-15T11:30:00+01:00"
}
```

Answer:

```
{ "changelog": [{"deletedWith": "", "modification": "CHANGE", "newValueClass": ""timeentry", "newValueDisplay": "Emma Clark, LEAVE {[2017-09-14T22:00:00.000Z]} - {[2017-09-15T10:30:00.000Z]}", "newValueId": "-1", "newValuesDictionary": false, "objectClass": "user", "objectId": 6, "objectLabel": "Emma Clark", "oldValueClass": "timeentry", "oldValueDisplay": "Emma Clark, LEAVE {[2017-09-14T22:00:00.000Z]} - {[2017-09-15T22:00:00.000Z]}", "oldValueId": "-1", "oldValuesDictionary": false, "property": "timeEntry"}]}
```

Create a new working time **with start and end time:**

<https://system.firmao.pl/{OrganizationID}/svc/v1/timeentries>

POST

```
{  
  
  "type":"TASK",  
  
  "user":1,  
  
  "dateTimeFrom":"2017-09-15T08:30:00+01:00",  
  
  "dateTimeTo":"2017-09-15T12:00:00+01:00"  
  
}
```

user - user identifier in the firmao system (number) to which we create the working timeAnswer:

```
{"changelog":[{"deletedWith":"","modification":"ATTACH","newValueClass":"timeentry","newValueDisplay":"Jacob Smith, TASK {[2017-09-15T07:30:00.000Z]} - {[2017-09-15T11:00:00.000Z]}", "newValueId":"50037", "newValueIsDictionary":false, "objectClass":"user", "objectId":1, "objectLabel":"Jacob Smith", "oldValueIsDictionary":false, "property":"timeEntry"}]}
```

7. PDF Download

a) Invoices

<https://system.firmao.pl/{identyfikacjaOrganizacji}>

}svc/v1/reports/invoices/transaction/**Faktura**{nazwa_pliku}.pdf?original=true&id=1

Where **Invoice** is the name of the template used, and File name is the name of the generated file.

b) Offers/orders/contracts

[https://system.firmao.pl/{idenfyfikatorOrganizacji}svc/v1/reports/invoices/offer/**Oferta**{nazwa_pliku}.pdf?id=1](https://system.firmao.pl/{idenfyfikatorOrganizacji}svc/v1/reports/invoices/offer/<u>Oferta</u>{nazwa_pliku}.pdf?id=1)

Where **Offer** is the name of the template used, and filename is the name of the generated file.

8. Uploading files/attachments to objects

a) (tasks/projects/products/services/customers/offers/orders/sales notes/opportunities)

POST :

https://system.firmao.pl/{idenfyfikatorOrganizacji}/svc/v1/{object}/{objectID}/documents/{filename}

```
{
  "description":"","
  "isFile":"true",
  "name":"invoice.pdf",
  "origin":"FILE",
  "size":96450],
  "url":""
}
```

Odpowiedź w przypadku poprawnego utworzenia Kod 201: {

```
"changelog": [  
  {  
    "deletedWith": "",  
    "modification": "CHANGE",  
    "newValueClass": "long",  
    "newValueDisplay": "1",  
    "newValuelsDictionary": false,  
    "objectClass": "customer",  
    "objectId": {objectId},  
    "objectLabel": {objectName},  
    "oldValueClass": "long",  
    "oldValueDisplay": "0",  
    "oldValuelsDictionary": false,  
    "property": "attachmentsCount"  
  }  
],  
"formAttributes": {  
  "enctype": "multipart/form-data",  
  "method": "POST",  
  "action": {properUrl}  
},  
"formFields": {  
  "signature": {signature},  
  "AWSAccessKeyId": {AWSAccessKeyId},  
  "acl": "private",  
  "success_action_redirect": {fileUrl},  
  "key": {key},  
  "policy": {policy}  
},  
"cookies": null,  
"maxFileSize": 779060875842  
}
```

9. Response Codes

API uses standard HTTP:2xx codes - error-free operation
4xx - an error occurred
5xx - an error occurred

Detailed description of some of the errors:

Valid request:200 - correct download or edit errors201 - correct creation of object

Bad request:400 - internal server error401 - request requires authorization403 - no access permissions404 - resource not found409 - conflict, operation canceled500 - unexpected error.501 - function not implemented502 - proxy

server errorIn the case of errors, a clear description of the error appears as standard.

For example:

```
{"errorCode":"entityNotFoundForId","field":"project","rejectedValue":"50","additionalInfo":  
{"id":"50","cls":"project"}}
```

10. Sample php scripts

NOTE: When copying code from a PDF, it often cuts off hyphens or adds

unnecessary newlines.

a) adding a contractor

```
<?php
```

```
$usr = base64_encode("login:password");
```

```
$url = 'https://system.firmao.pl/{OrganizationID}/svc/v1/customers';
```

```
$content = array(
```

```
"bankAccountNumber" => "50 1020 5558 1111 1702 0600 0067",
```

```
"correspondenceAddress.city" => "Pruszków",
```

```
"correspondenceAddress.country" => "Poland",
```

```
"correspondenceAddress.postCode" => "05-800",
```

```
"correspondenceAddress.street" => "Stefana Batorego 10/4",
```

```
"customerGroups" => array(),
```

```
"customerType" => "PARTNER",
```

```
"description" => "A company specializing in computer service",
```

```
"emails" => array("kompservice.firmao@mailinator.com", "komputery@mailinator.com"),
```

```
"employeesNumber" => "1",
```

```
"faxNumber" => "22 398 78 87",  
  
"label" => "KOMP-SERVICE",  
  
"name" => "KOMP-SERVICE Jan Kowalski",  
  
"nipNumber" => "7291077843",  
  
"officeAddress.city" => "Pruszków",  
  
"officeAddress.country" => "United States",  
  
"officeAddress.postCode" => "05-800",  
  
"officeAddress.street" => "Stefana Batorego 3/4",  
  
"ownership" => "PRIVATE",  
  
"phones" => "223457426",  
  
"status" => null,  
  
"tags" => array(),  
  
"voivodeship" => null,  
  
"website" => "www.kompservice.bis");
```

```
$json = json_encode($content);
```

```
echo $json.' <br /><nr />';
```

```
$curl = curl_init($url);
```

```
curl_setopt($curl, CURLOPT_CUSTOMREQUEST, "POST");

curl_setopt($curl, CURLOPT_HEADER, false);

curl_setopt($curl, CURLOPT_RETURNTRANSFER, true);

curl_setopt($curl, CURLOPT_HTTPHEADER,

array("Content-type: application/json; charset=UTF-8", "Authorization: Basic $usr"));

curl_setopt($curl, CURLOPT_POSTFIELDS, $json);
```

accepting curl_setopt certificates

```
($curl, CURLOPT_SSL_VERIFYHOST, 0);
```

```
curl_setopt ($curl, CURLOPT_SSL_VERIFYPEER, 0);
```

```
$json_response = curl_exec($curl);
```

```
$status = curl_getinfo($curl, CURLINFO_HTTP_CODE);
```

```
if ( $status != 201 && $status != 200) {
```

```
    die("Error: call to URL $url failed with status $status, response $json_response,
    curl_error " . curl_error($curl) . ", curl_errno " . curl_errno($curl));
```

```
}
```

```
curl_close($curl);
```

```
$response = json_decode($json_response, true);
```

```
?>
```

b) downloading the contractor's data

```
<?php
```

```
$usr = base64_encode("login:password");
```

```
$url = 'https://system.firmao.pl/{OrganizationID}/svc/v1/customers';
```

```
$curl = curl_init($url);
```

```
curl_setopt($curl, CURLOPT_CUSTOMREQUEST, "GET");
```

```
curl_setopt($curl, CURLOPT_HEADER, false);
```

```
curl_setopt($curl, CURLOPT_RETURNTRANSFER, true);
```

```
curl_setopt($curl, CURLOPT_HTTPHEADER,
```

```
array("Content-type: application/json; charset=UTF-8", "Authorization: Basic $usr"));
```

```
curl_setopt($curl, CURLOPT_POSTFIELDS, $json);
```

Accepting certificates

```
curl_setopt ($curl, CURLOPT_SSL_VERIFYHOST, 0);
```

```
curl_setopt ($curl, CURLOPT_SSL_VERIFYPEER, 0);

$json_response = curl_exec($curl);

$status = curl_getinfo($curl, CURLINFO_HTTP_CODE);

if ( $status != 201 && $status != 200) {

die("Error: call to URL $url failed with status $status, response $json_response,

curl_error " . curl_error($curl) . ", curl_errno ". curl_errno($curl));

}

curl_close($curl);

echo $json_response;

$response = json_decode($json_response, true);

?>
```

c) adding transactions

```
<?php
```

```
$usr = base64_encode("login:password");
```

```
$url = 'https://system.firmao.pl/{OrganizationID}/svc/v1/transactions';
```

```
$content = array(
"transactionEntries"=>array(array(
    "classificationNumber"=>"",
    "discount"=>null,
    "productCode"=>"ST0001",
    "quantity"=>1,
    "unit"=>null,
    "vatPercent"=>0,
    "product"=>51,
    "unitNettoPrice"=>"100.00"
)),
"transactionNettoPrice"=>"1000.00",
"transactionBruttoPrice"=>"1000.00",
"transactionVatPrice"=>"0.00",
"invoiceAnnotations"=>null,
"paid"=>false,
"currency"=>"PLN",
"paymentType"=>"CASH",
```

```
"restOfPaidValue"=>1000,  
  
"paidValue"=>0,  
  
"invoiceDate"=>"2012-08-08",  
  
"paymentDate"=>"2012-08-08",  
  
"transactionDate"=>"2012-08-08",  
  
"type"=>"INVOICE",  
  
"mode"=>"SALE",  
  
"customerAddress.postCode"=>"91477",  
  
"customer"=>50,  
  
"customerAddress.city"=>"Chicago",  
  
"customerAddress.country"=>"United States",  
  
"customerAddress.street"=>"Piotrkowska 147/8",  
  
"nipNumber"=>"1060000062",  
  
"issuingPerson"=>"Jacob Smith",  
  
"transactionNumber"=>"12/8/2012"  
  
);  
  
$json = json_encode($content);  
  
echo $json.' <br /><nr />';  
  
$curl = curl_init($url);
```

```
curl_setopt($curl, CURLOPT_CUSTOMREQUEST, "POST");

curl_setopt($curl, CURLOPT_HEADER, false);

curl_setopt($curl, CURLOPT_RETURNTRANSFER, true);

curl_setopt($curl, CURLOPT_HTTPHEADER, array("Content-type:application/json;
charset=UTF8", "Authorization: Basic $usr"));

curl_setopt($curl, CURLOPT_POSTFIELDS, $json);
```

Accepting certificates

```
curl_setopt ($curl, CURLOPT_SSL_VERIFYHOST, 0);

curl_setopt ($curl, CURLOPT_SSL_VERIFYPEER, 0);

$json_response = curl_exec($curl);

$status = curl_getinfo($curl, CURLINFO_HTTP_CODE);

if ( $status != 201 && $status != 200) {

die("Error: call to URL $url failed with status $status, response $json_response,

curl_error " . curl_error($curl) . ", curl_errno ". curl_errno($curl));

}

curl_close($curl);

$response = json_decode($json_response, true);

?>
```

d) Uploading the PHP file < 7.0

```
<?php
```

```
$usr = base64_encode("login:password");
```

```
$url = 'https://system.firmao.pl/{OrganizationID}/svc/v1/{object}/{id}/documentsUpload';
```

```
    $curl = curl_init($url);
```

```
    $parameters = array(
```

```
        'description'=>date('Y-m-d'),
```

```
        'file'=>'@.' c:\test.txt');
```

```
    curl_setopt($curl, CURLOPT_CUSTOMREQUEST, "POST");
```

```
    curl_setopt($curl, CURLOPT_POSTFIELDS, $parameters);
```

```
    curl_setopt($curl, CURLOPT_HEADER, false);
```

```
    curl_setopt($curl, CURLOPT_RETURNTRANSFER, true);
```

```
    curl_setopt($curl,
```

```
    CURLOPT_HTTPHEADER,array("Content-type:multipart/form-data;  
charset=UTF8","Authorization: Basic $usr"));
```

```
curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, 0);

curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, 0);

$json_response = curl_exec($curl);

$status = curl_getinfo($curl, CURLINFO_HTTP_CODE);

if ($status != 201 && $status != 200) {

    die("Error: call to URL $url failed with status $status, response $json_response,
curl_error " . curl_error(

        $curl

    ) . ", curl_errno ". curl_errno($curl));

}

curl_close($curl);

$response = json_decode($json_response, true);

?>
```

e) uploading php file >= 7.0

```
<?php
```

```
$usr = base64_encode("login:password");
```

```
$url = 'https://system.firmao.pl/{OrganizationID}/svc/v1/{object}/{id}/documentsUpload';
```

```
    $curl = curl_init($url);
```

```
$parameters = array(
```

```
    'file' => new CURLFile("$filePath")
```

```
);
```

```
curl_setopt($curl, CURLOPT_CUSTOMREQUEST, 'POST');
```

```
curl_setopt($curl, CURLOPT_HEADER, false);
```

```
curl_setopt($curl, CURLOPT_RETURNTRANSFER, true);
```

```
curl_setopt($curl, CURLOPT_HTTPHEADER, array("Authorization: Basic $usr"));
```

```
curl_setopt($curl, CURLOPT_POSTFIELDS, $data);
```

```
curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, 0);
```

```
curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, 0);
```

```
$json_response = curl_exec($curl);
```

```
$status = curl_getinfo($curl, CURLINFO_HTTP_CODE);

if ($status != 201 && $status != 200) {

    die("Error: call to URL $url failed with status $status, response $json_response,
curl_error " . curl_error(

        $curl

    ) . ", curl_errno ". curl_errno($curl));

}

curl_close($curl);

$response = json_decode($json_response, true);

?>
```

11. More API Options

If you need more API options or have problems with configuration, please write to us at kontakt@firmao.pl